BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

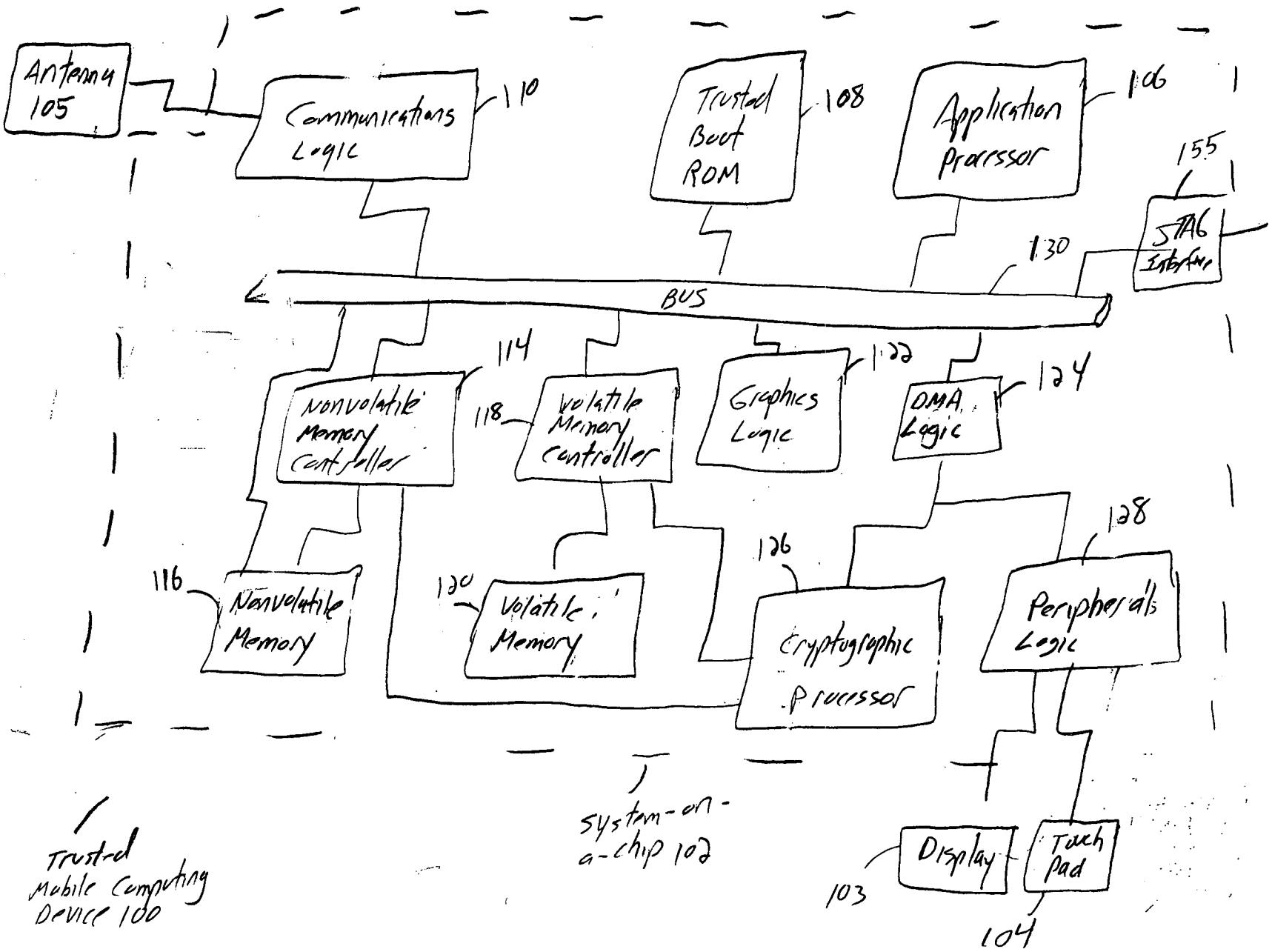
Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

FIG. 1


Antenna 105
Communications Logic
110
Trusted Boot ROM
108
Application Processor
106
155
JTAG Interface
130
BUS
Nonvolatile Memory Controller
114
Volatile Memory Controller
118
Graphics Logic
122
DMA Logic
124
Nonvolatile Memory
116
Volatile Memory
120
Cryptographic Processor
126
Peripherals Logic
128
Display
103
Touch Pad
104
System-on-a-chip 102
Trusted Mobile Computing Device 100

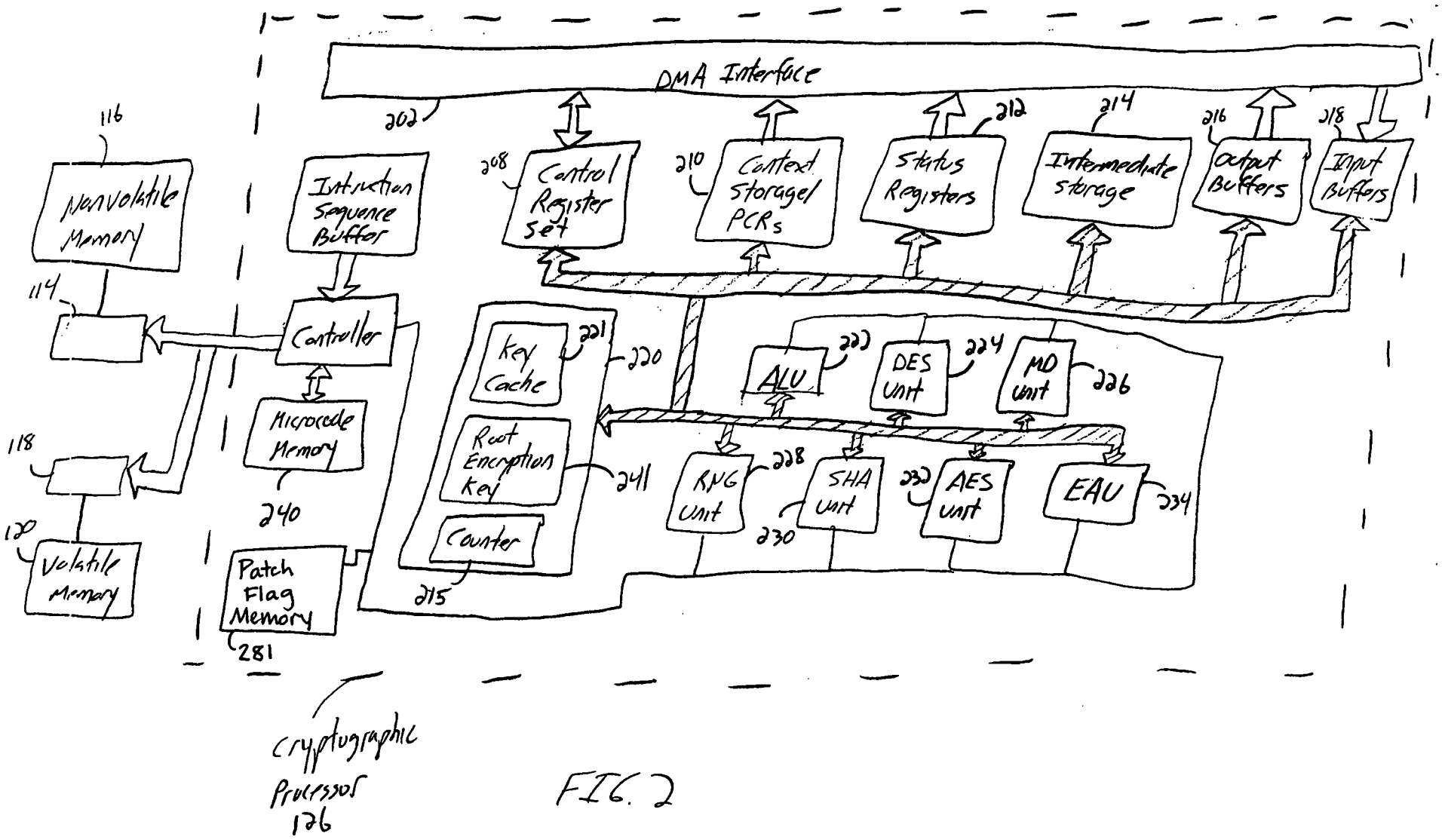

DMA Interface
202
116
Nonvolatile Memory
114
Instruction Sequence Buffer
Controller
208
Control Register Set
210
Context Storage/ PCRs
212
Status Registers
214
Intermediate storage
216
Output Buffers
218
Input Buffers
118
Microcode Memory
240
120
Volatile Memory
Patch Flag Memory
281
Key Cache
221
220
Root Encryption Key
241
Counter
215
ALU
222
DES Unit
224
MD Unit
226
RNG Unit
228
SHA Unit
230
AES Unit
232
EAU
234
Cryptographic Processor 126

FIG. 2

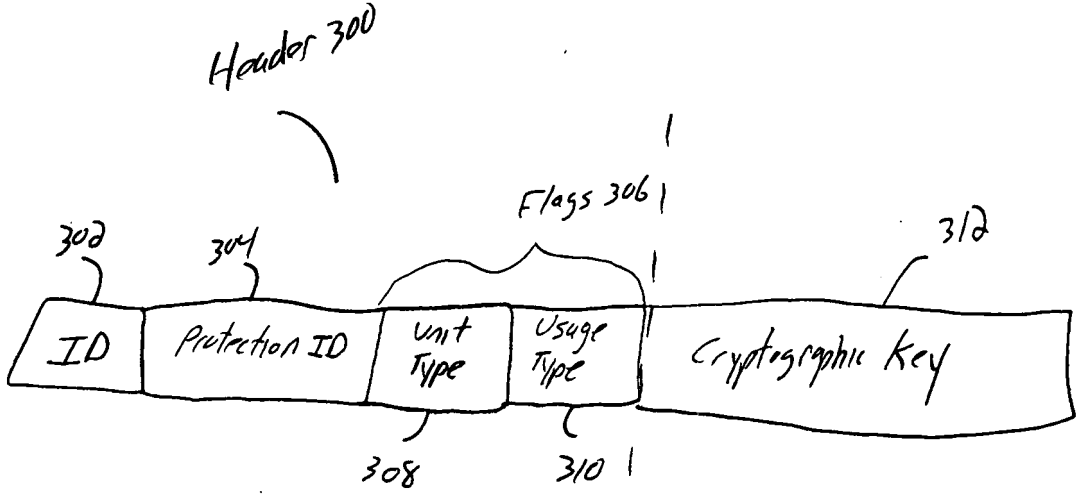

Header 300
Flags 306
302
304
312
ID
Protection ID
Unit Type
Usage Type
Cryptographic Key
308
310

FIG. 3

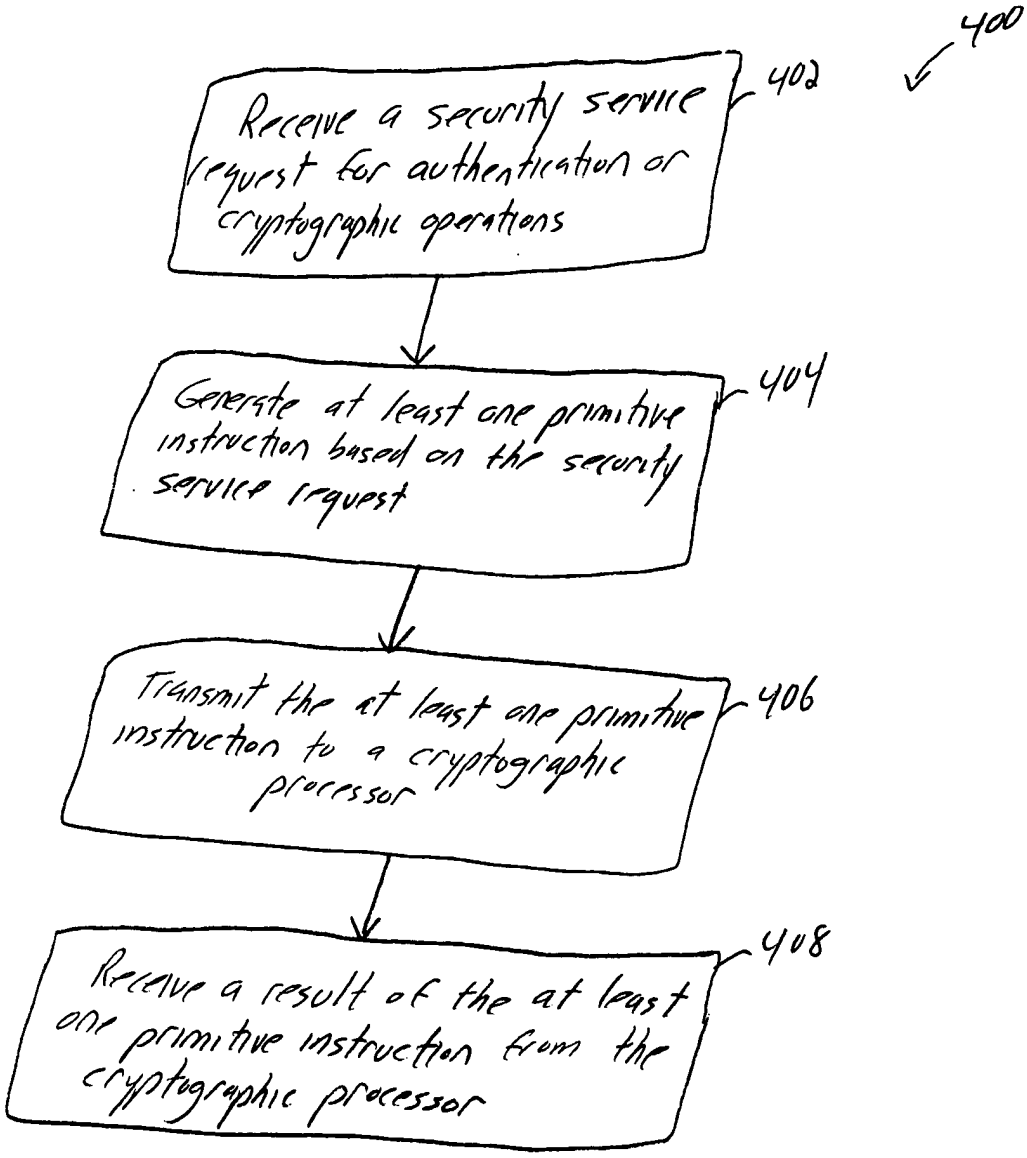

400
Receive a security service request for authentication or cryptographic operations
402
Generate at least one primitive instruction based on the security service request
404
Transmit the at least one primitive instruction to a cryptographic processor
406
Receive a result of the at least one primitive instruction from the cryptographic processor
408

FIG. 4

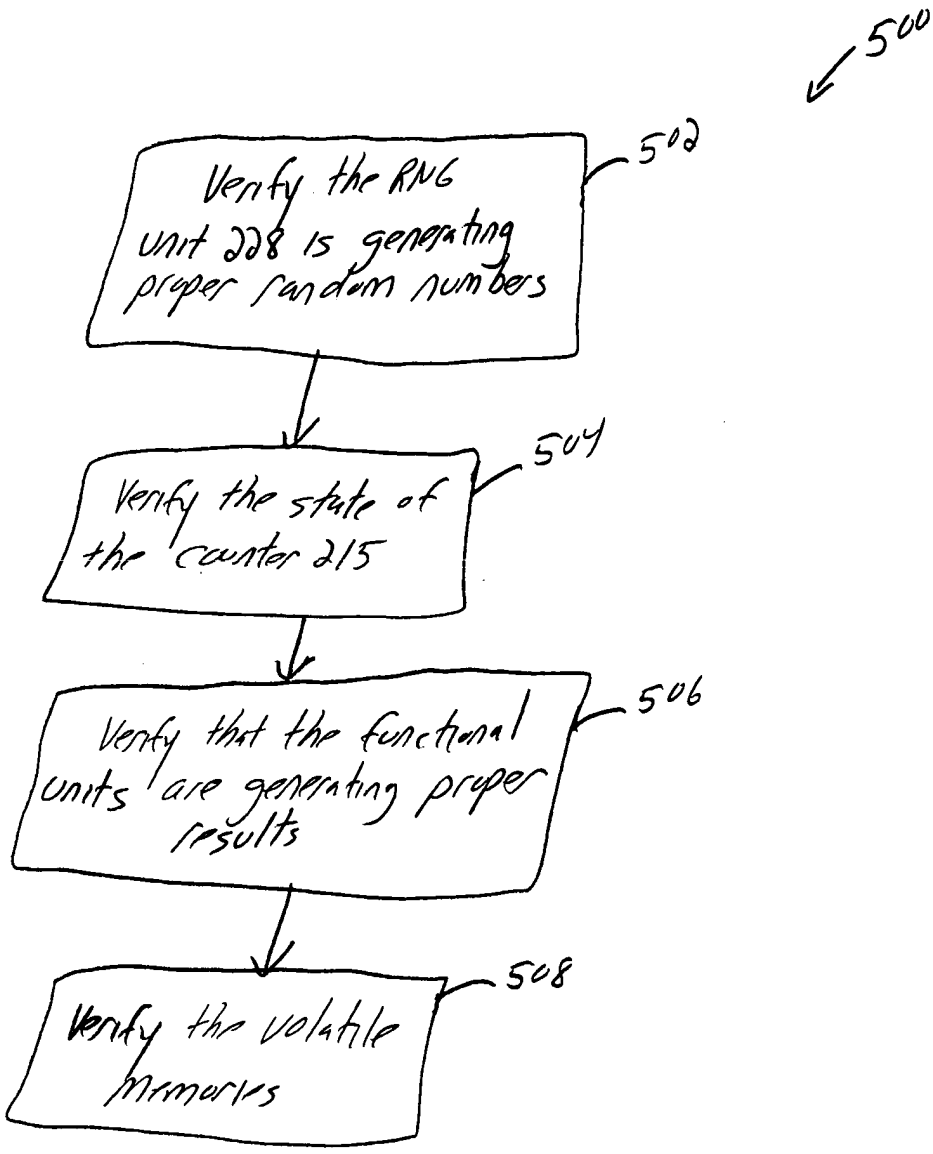

500
Verify the RNG unit 228 is generating proper random numbers
502
Verify the state of the counter 215
504
Verify that the functional units are generating proper results
506
Verify the volatile memories
508

FIG. 5

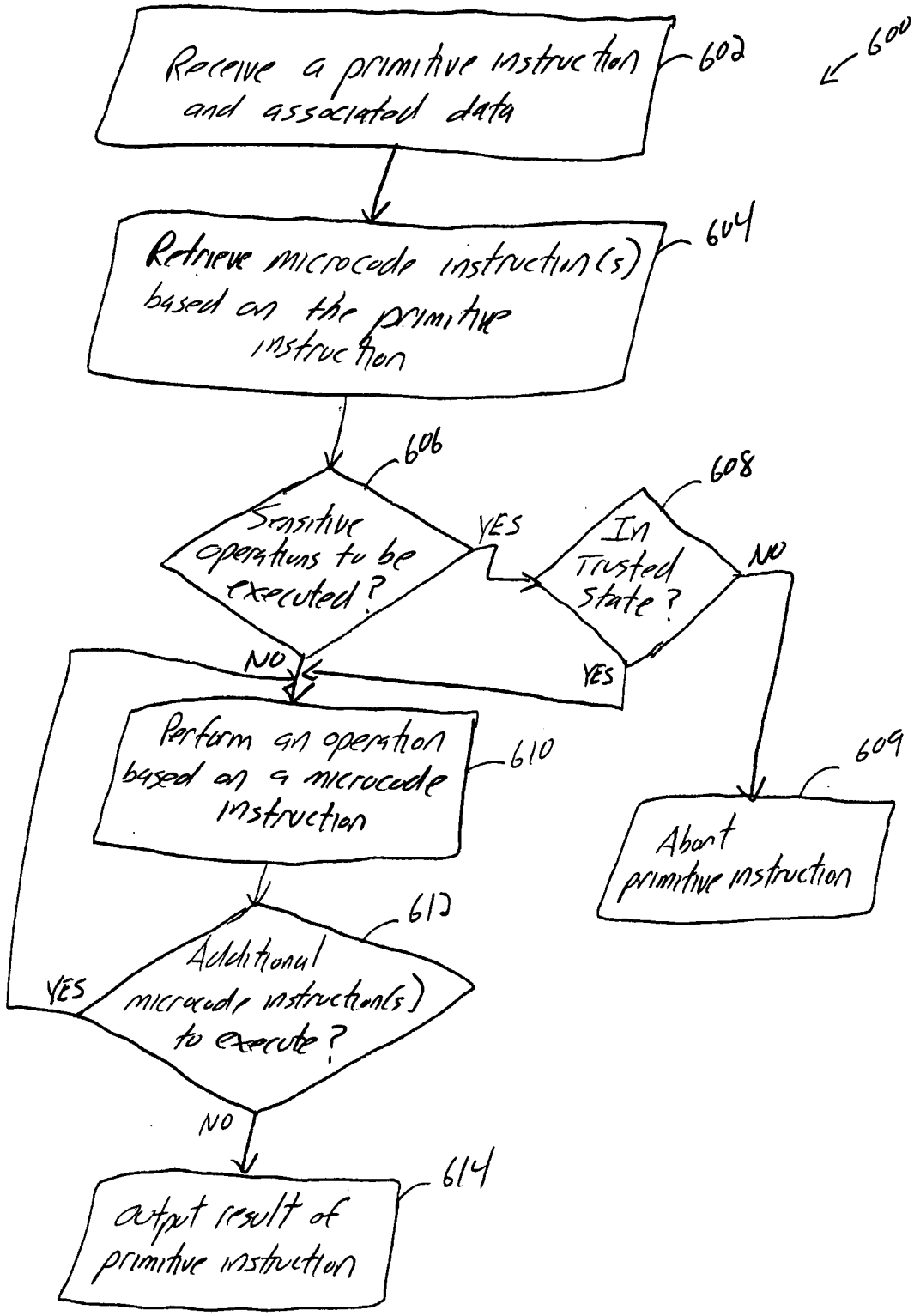

Receive a primitive instruction and associated data
602
600
Retrieve microcode instruction(s) based on the primitive instruction
604
606
Sensitive operations to be executed?
YES
608
In Trusted State?
NO
NO
YES
Perform an operation based on a microcode instruction
610
609
Abort primitive instruction
612
Additional microcode instruction(s) to execute?
YES
NO
614
output result of primitive instruction

FIG. 6A

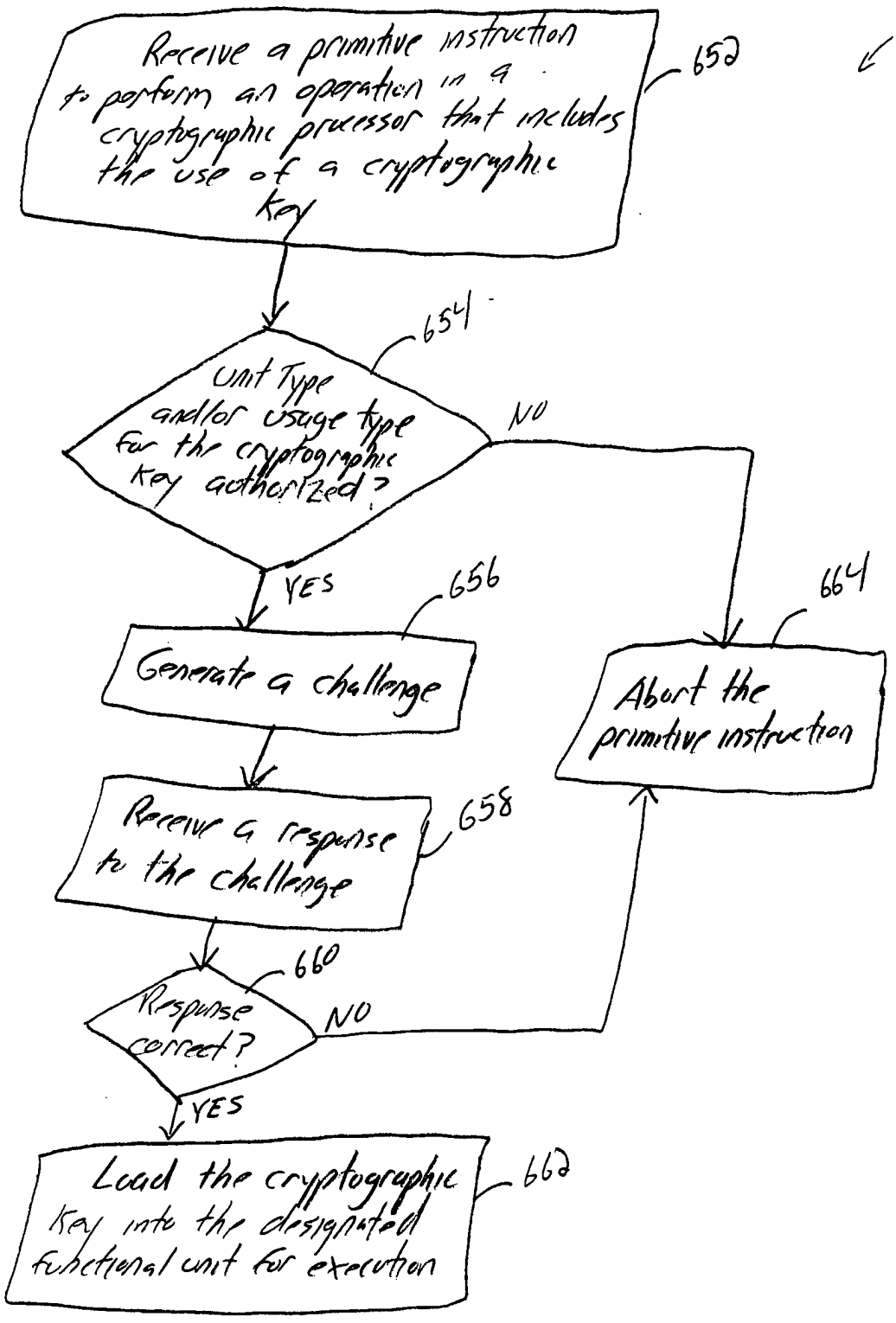

650
Receive a primitive instruction to perform an operation in a cryptographic processor that includes the use of a cryptographic key
652
654
Unit Type and/or usage type for the cryptographic key authorized?
NO
YES
656
Generate a challenge
658
Receive a response to the challenge
660
Response correct?
NO
YES
662
Load the cryptographic key into the designated functional unit for execution
664
Abort the primitive instruction

FIG. 6B

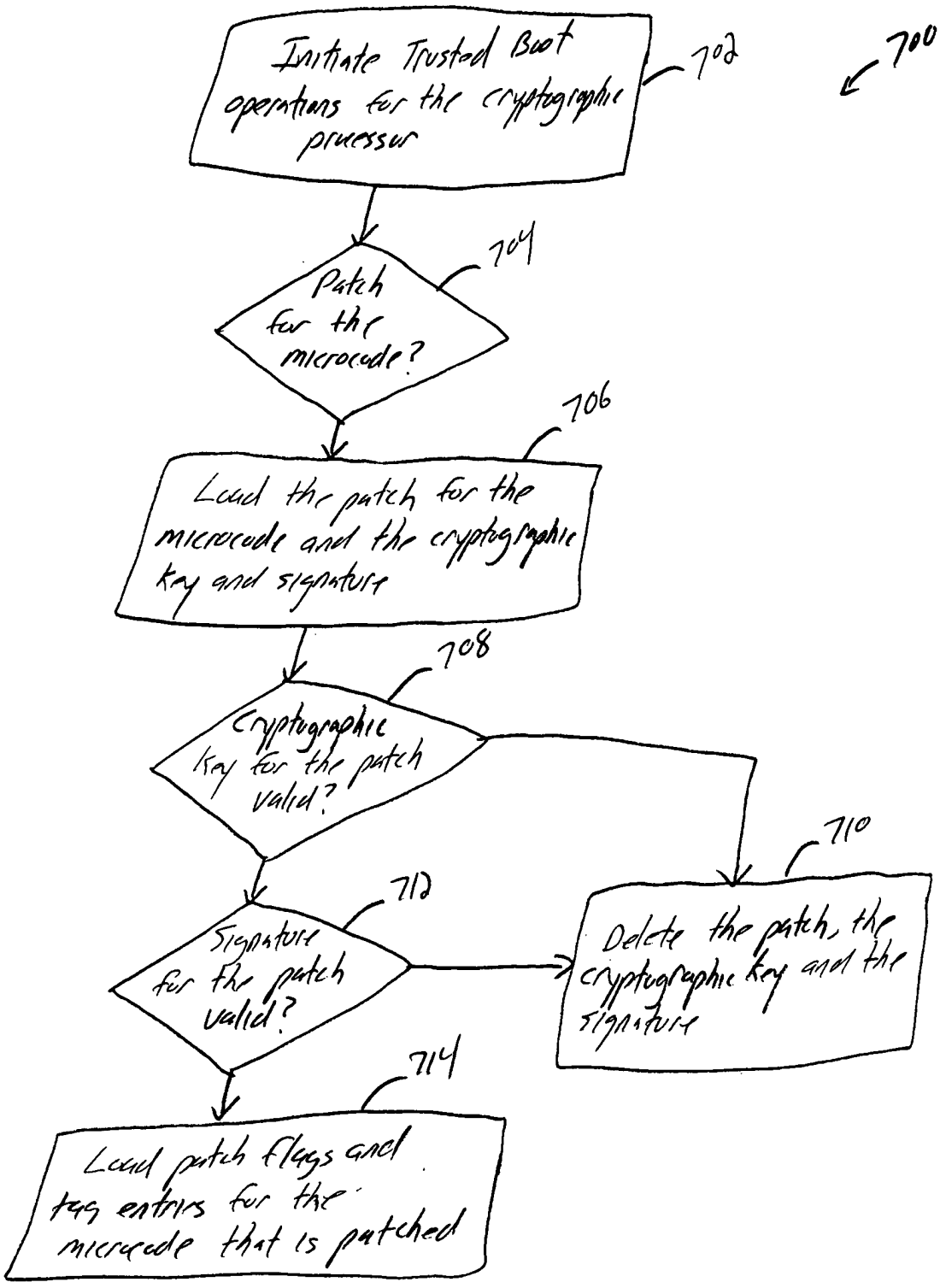

Initiate Trusted Boot operations for the cryptographic processor
702
700
Patch for the microcode?
704
Load the patch for the microcode and the cryptographic key and signature
706
Cryptographic key for the patch valid?
708
Signature for the patch valid?
712
Delete the patch, the cryptographic key and the signature
710
Load patch flags and tag entries for the microcode that is patched
714

FIG 7

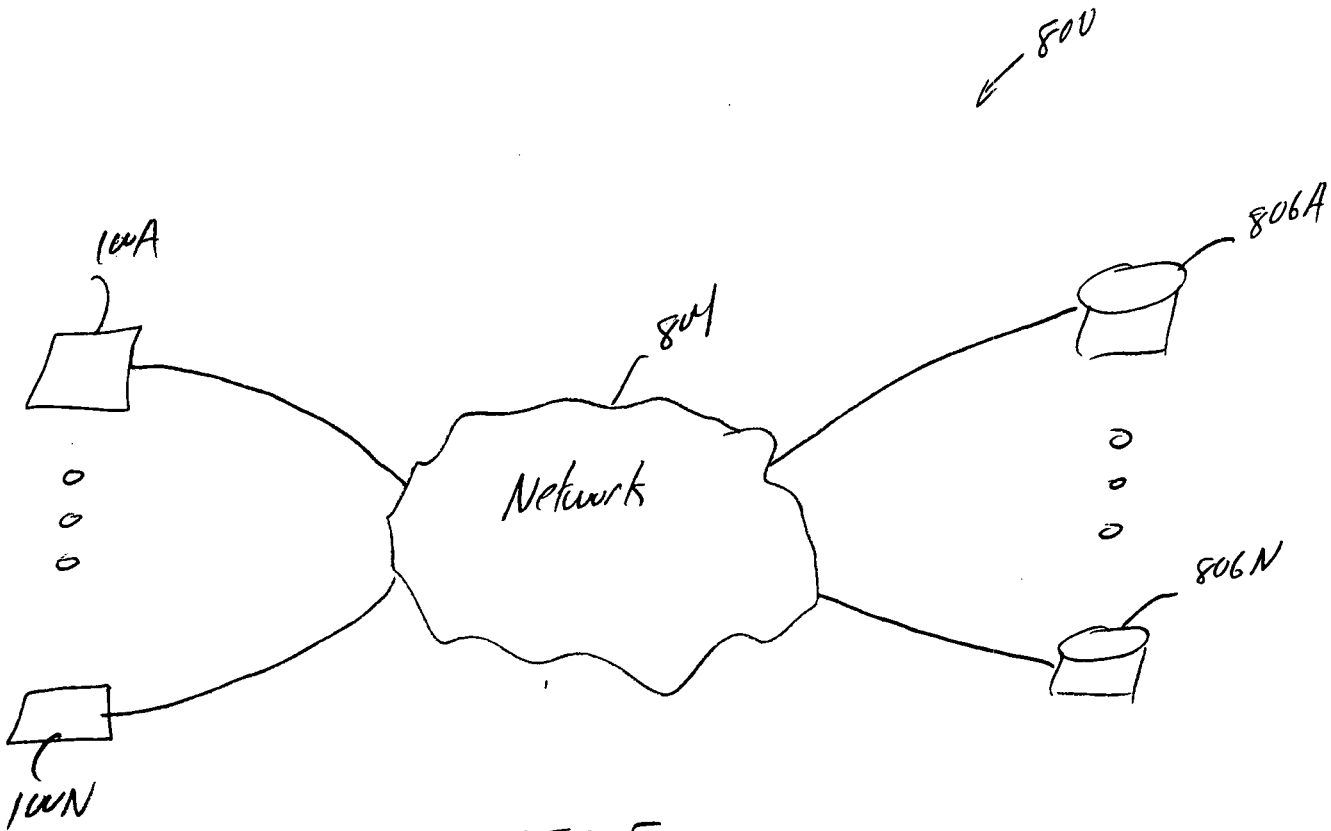


FIG. 8